OPTIMISATION PAR COLONIE DE FOURMIS

Intelligence en Essaim

PhilippeCollard.com

Optimisation par colonie de fourmis

- Ethologie
- Optimisation par Colonie de fourmis
- Ant System
- Problème du voyageur de commerce
- Modélisation & Simulation en NetLogo

Ethologie vs. Informatique

- Ethologie
  - Comportements collectifs des insectes sociaux
- Informatique
  - Traitements et données distribuées
  - Données situées (notion d'espace)
  - Communication temps/espace (Stigmergie)
  - méthodes puissantes pour la conception d'algorithmes
    - Optimisation combinatoire
    - Routage
    - Contrôle distribué

Ethologie vs. Informatique

- Les problèmes quotidiens résolus par une colonie sont nombreux et de nature très variée
  - recherche de nourriture
  - construction du nid
  - division du travail
  - allocation des tâches entre les individus, etc

- La plupart de ces problèmes se retrouvent dans le domaine des sciences de l'ingénieur, en informatique et en robotique

Méthodes flexibles et robustes

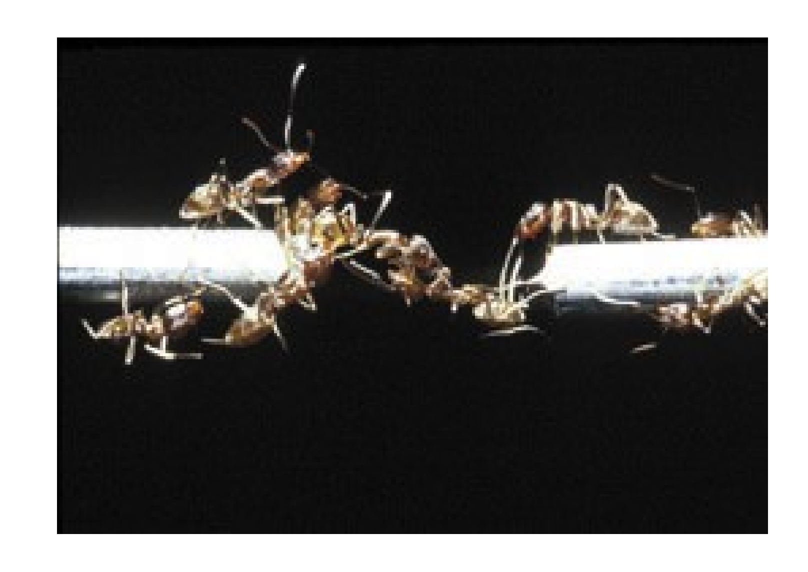
- En plus de leur capacité, déjà surprenante, à résoudre un large spectre de problèmes *statiques*, ces techniques offrent un haut degré de flexibilité et de robustesse dans des environnements *dynamiques*

- Elles permettent de résoudre de façon plus efficace des problèmes d'optimisation, comme les problèmes
  - d'assignation quadratique (QAP)
  - d'adaptation de flux des communications circulant sur un réseau

Les sociétés d'insectes ont une capacité à résoudre des problèmes d'une manière …

- Flexible
  - la colonie s'adapte aux brusques changements d'environnement
- Robuste
  - la colonie continue de fonctionner lorsque certains individus échouent à accomplir leur tâche

Comportements collectifs des insectes sociaux auto-organisés





Comportements collectifs Auto-Organisés

- L'auto-organisation = <u>émergence</u> de structures au niveau collectif, à partir d'une multitude d'interactions simples, sans être codées explicitement au niveau individuel

- Certaines interactions - *une fourmi qui suit la piste de phéromone laissée par une autre* - aident à résoudre collectivement des problèmes difficiles, par exemple trouver le chemin le plus court conduisant à une source de nourriture

Intelligence en essaim :
un nouveau domaine de recherche

- Transformer la connaissance que les éthologistes ont des capacités collectives de résolution de problèmes des insectes sociaux en techniques artificielles de résolution de problèmes

- Les informaticiens et les ingénieurs ont pu transformer des modèles du comportement collectif des insectes sociaux en méthodes utiles pour l'optimisation et le contrôle

Intelligence en essaim
un nouveau domaine de recherche

- Parmi les techniques de l'intelligence en essaim, certaines sont arrivées à maturité

- Les algorithmes de contrôle et d'optimisation inspirés de modèles de recherche collective de nourriture chez les fourmis en particulier, ont connu un succès inattendu et portent le nom d'optimisation par colonie de fourmis (Ant Colony Optimization) et de "routage par colonie de fourmis"

Optimisation par colonie de fourmis

- Les fourmis sont capables de sélectionner le plus court chemin pour aller du nid à une source de nourriture grâce au dépôt et au suivi de pistes de phéromone

- Lorsqu'une colonie de fourmis d'Argentine doit emprunter un pont à deux branches de longueurs différentes pour exploiter une source de nourriture, elle sélectionne la branche courte si la différence entre les longueurs des branches est suffisamment importante

A.C.O
comment ça marche ?

- Les fourmis déposent de la phéromone à l'aller vers la source de nourriture et au retour vers le nid

- Au départ, le choix est aléatoire mais la branche courte devient vite la plus marquée car les fourmis qui l'empruntent arrivent plus vite au nid et auront statistiquement plus de chance de l'emprunter lorsqu'elles retourneront vers la source de nourriture

Phéromone

- Les fourmis choisissent la piste qui porte la plus forte concentration
- Piste chimique virtuelle

Important de comprendre les comportements naturels avant d'abstraire dans un algorithme

Phéromone ?

- Substance chimique produite par des glandes déclenchant des réactions comportementales entre individus de la même espèce
- Signaux chimiques odorants agissant à grande distance à dose moléculaire
- Moyen de communication chez les insectes
- Plusieurs types : sexuelles, de piste, grégaires, d'alarme,…

Phéromone ?

- Chimiorécepteurs = antennes garnies d'organes sensoriels

- Applications pratiques en agriculture
  - "pièges à phéromones " composés d'un attractif (synthèse de la phéromone naturelle de la femelle du ravageur) et d'un système assurant la capture des mâles

Mouche de l'olive : piège à phéromones

- Piège composé d'un toit englué et d'une capsule de phéromone suspendue au milieu

- Mode d'emploi : dans le cas général, il faut 1 piège par ha



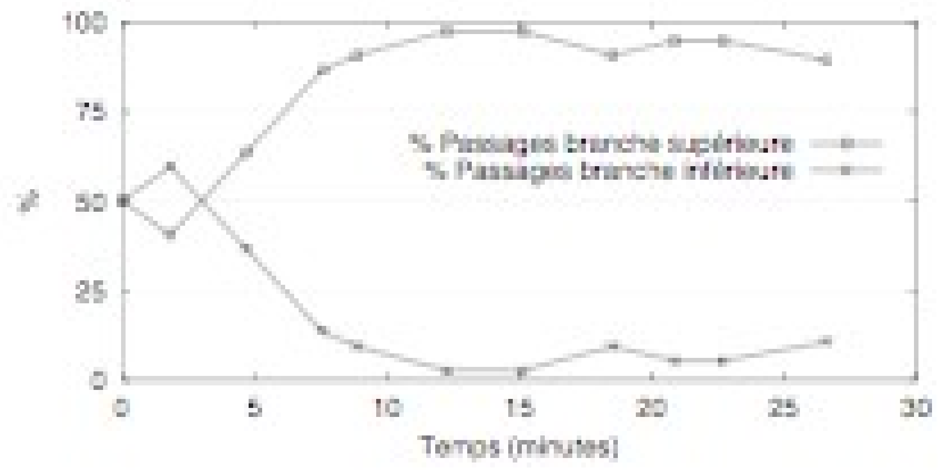
- http://www.biotop.fr/04produits/mod_pheromon.htm

Phénomène autocatalytique

Pont binaire de Deneubourg (1999)

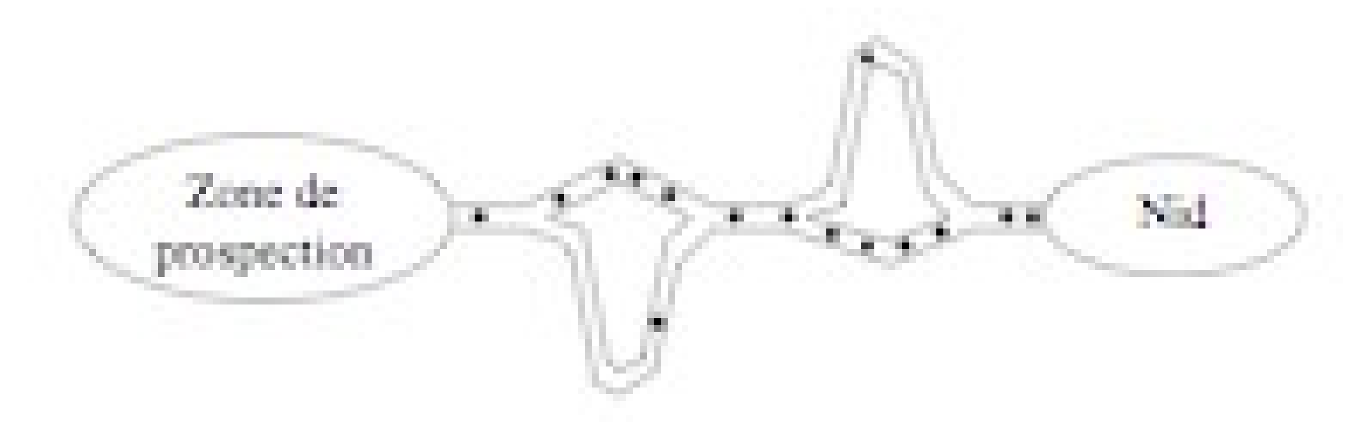
- Initialement le pont est vide
- Apres une période transitoire, des fluctuations aléatoires favorisent la piste supérieure … plus les A suivent une piste, plus elle devient attractive
- Phénomène qui se renforce lui-même (positive feedback)



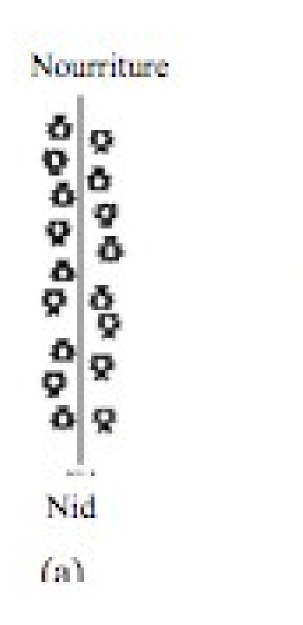


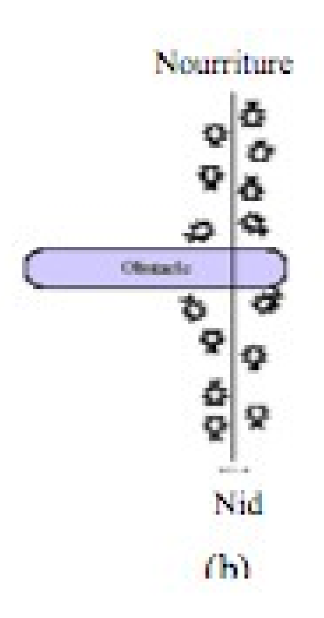
Expérience du double pont binaire

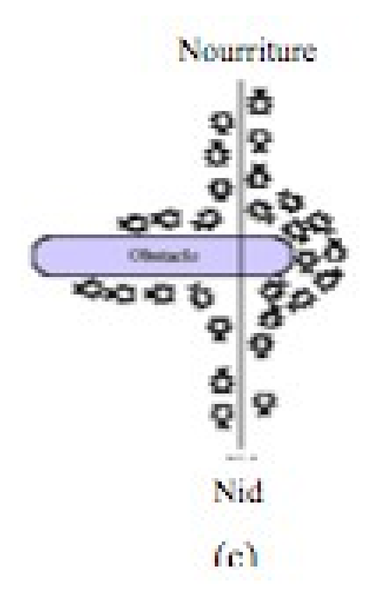
- Influence des fluctuations aléatoires initiales réduite
- Car les 4 chemins possibles sont de longueurs différentes



Effet de la coupure d'une piste







Stigmergie

- Stimulation d'agents par la performance de ce qu'ils ont accompli [Grassé 1999]
  - Communication indirecte
  - Pas de contact physique entre agents
  - Pas de centralisation : interaction locale
- Optimisation = Propriété émergente

Ant Colony Optmisation

- Introduit par M. Dorigo en 1991
- Historiquement appliqué sur le TSP

- Idée : représenter le problème à résoudre sous la forme de la recherche d'un meilleur chemin dans un graphe
  - <u>Intensification</u> (via les phéromones) des « bons » chemins
  - <u>Diversification</u> par la nature aléatoire des décisions

- Méta-problème : gérer le compromis exploitation/exploration

A.C.O et T.S.P

- Le problème du voyageur de commerce (Traveller Salesman Problem) consiste à trouver le chemin le plus court en passant une seule fois par un nombre donné de villes
- En utilisant des fourmis artificielles conçues pour déposer des pistes de phéromone dont la concentration varie en fonction de la distance totale qu'elles ont parcourue, on peut obtenir des chemins quasi optimaux

Problème du Voyageur de Commerce (TSP)

- Complexité : # de parcours possibles, pour n villes est de (n-1)!
  - 6 villes, 120 parcours possibles
  - 10 villes plus de 362 000
  - 60 villes plus de 10^{80} (# atomes dans l'univers)

- Ceci peut expliquer pourquoi, le problème n'a pas été étudié sérieusement avant l'arrivée des ordinateurs mais depuis de nombreux chercheurs l'ont traité

Ant System

- Le premier ACO (1992)
  - Performances très moyennes
  - Etendue à de nouvelles versions : `Max-Min AS`

- Robuste

  la colonie continue de fonctionner lorsque certains individus échouent à accomplir leur tâche

Algorithme **AS**

- Initialiser
- **repeat**
  - Chaque ant construit une solution (un tour)
  - [améliorer les solutions par recherche locale]
  - « Récompenser » les meilleures solutions en ajoutant de la phéromone
  - « Evaporer » les traces de phéromone

Until (maxCycles ou bonneSolution)

AntSystem : initialisation

1. Les `m` ants sont réparties aléatoirement sur les `n` villes

2. Pour chaque ant la `liste-tabou` contient sa ville de départ

3. Les pistes de phéromones sont initialisées : $\tau_{ij} = c$, `c` constant positive non nulle

AntSystem : itération de base

Chaque ant choisit une ville de destination et s'y déplace

Une ant k placée sur une ville i à l'instant t choisit sa ville de destination j en fonction de la :

1. visibilité de cette ville η_{ij} (distance inter villes)
2. quantité de phéromone $\tau_{ij}(t)$ déposée sur l'arc reliant ces deux villes

- Le choix est aléatoire selon une probabilité où deux paramètres α et β contrôlent l'importance relative des phéromones et de la visibilité

AntSystem : itération de base

Chaque ant choisit une ville de destination et s'y déplace

Le choix est aléatoire selon une probabilité où deux paramètres α et β contrôlent l'importance relative des phéromones et de la visibilité

La probabilité d'un déplacement élémentaire est un compromis entre visibilité et piste chimique

- Si $\beta=0$, les villes les + proches ont + de chance d'être sélectionnées (algorithme glouton)
- Si $\alpha=0$, seule l'amplification des phéromones agit : convergence prématurée

Fin d'un cycle de base : Toutes les ants ont terminé un tour en revenant à leur propre ville de départ

- Pour chaque ant k :
  - calculer la longueur de son tour $L_k(t)$
  - vider sa liste-tabou
- mettre à jour les phéromones τ_{ij}
- rechercher le plus petit tour et le mémoriser s'il est meilleur que les précédents
- Toutes les ants recommencent un nouveau tour à partir de leur propre ville initiale

Mettre à jour les phéromones

$$\tau_{ij}(t+n) = \rho . t_{ij}(t) + \Delta\tau_{ij}(t)$$

ρ dans $[0,1[$

Entre les instants t et $t+n$

- $(1-\rho)$ = évaporation de la piste
- $\Delta\tau_{ij}(t)$ = quantité de phéromone déposée par les ants

Evaporation des phéromones

- Si $\rho=1$, pas d'évaporation donc pas de limitation du phénomène autocatalytique

- Si $\rho=0$, les ants prennent seulement en compte les dépôts du dernier cycle

- ρ représente la persistance de la piste (ie. l'effet mémoire)

Quantité de phéromone déposée par les ants lors d'un tour (cycle)

$\Delta\tau_{ij}(t)$ = quantité de phéromone déposée par les ants sur l'arc reliant la ville i à la ville j entre les instants t et $t+n$

Pour chaque ant k passant par l'arc (i,j),

$$\Delta\tau_{ij}(t) \mathrel{+}= Q/L_k(t)$$

où Q représente un « quota » de phéromones attribué à chaque ant (souvent Q=100)

Idée : + un tour est court, + les arcs qui le composent sont approvisionnés

Remarque : c'est une m.a.j « retardée »

Algorithme **Max-MinAS**

- Initialiser
- **repeat**
  - Chaque ant construit une solution (un tour)
  - [améliorer les solutions par recherche locale]
  - « Récompenser » les meilleurs solutions en ajoutant de la phéromone
  - « Evaporer » les traces de phéromone
  - **Si une tracePhéromone<τ_{min} alors la mettre à τ_{min}**
  - **Si une tracePhéromone>τ_{max} alors la mettre à τ_{max}**

Until (maxCycles ou bonneSolution)

Algorithme **Max-MinAS**

- Fournir des résultats compétifs
- Imposer des bornes τ_{min} et τ_{max} aux traces de phéromones
- Les traces sont initialisés avec τ_{max}

Quel est l'effet de ces choix sur le fonctionnement de l'algorithme ?

A voir en Travaux dirigés …

Optimisation par colonie de fourmis et TSP

	Oliver30 (30-city)	Eil50 (50-city)	Eil75 (75-city)
OCF	420 (830)	425 (1830)	535 (3480)
Genetic algorithm	421 (3200)	428 (25000)	545 (80000)
Evolutionary programming	420 (40000)	426 (100000)	542 (325000)
Simulated annealing	424 (24617)	443 (68512)	580 (173250)
Optimal solution	420	425	535

Cette optimisation est une conséquence de l'interaction subtile entre <u>renforcement</u> et <u>évaporation</u> de la phéromone, qui fait que seules les meilleures liaisons subsistent

Software package ACOTSP.V1.0

- Auteur : Thomas Stützle
- ACOs appliqués aux TSP symétriques
  - Ant System (AS)
  - Elitist Ant System (EAS)
  - MAX-MIN Ant System (MMAS)
  - Rank-based version of Ant System (RAS)
  - Best-Worst Ant System (BWAS)
  - Ant Colony System (ACS)
- ANSI C sous Linux (works fine under Windows and Mac OS X)

ACOTSP : command line options (1)


```
-r          # number of independent trials
-s          # number of steps in each trial
-t          # maximum time for each trial
-i          f inputfile (TSPLIB format necessary)
-o          # stop if tour >= optimum is found
-m          # number of ants
-g,         # nearest neighbours in tour construction
-a          # alpha (influence of pheromone trails)
-b          # beta (influence of heuristic information)
-e          # rho: pheromone trail evaporation
-q          # q_0: prob. of best choice in tour construction
-c          # number of elitist ants
-f          # number of ranks in rank-based Ant System
-k          # No. of nearest neighbors for local search
-l      0: no local search   1: 2-opt   2: 2.5-opt   3: 3-opt
-d          1 use don't look bits in local search
```

ACOTSP : command line options (2)

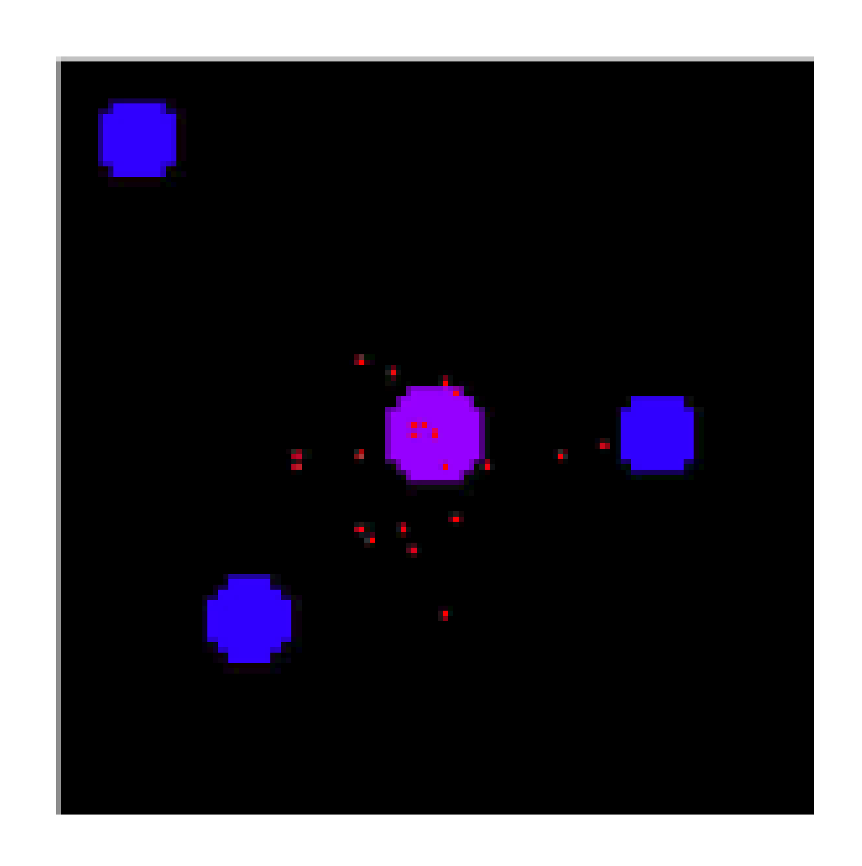

```
-u, --as     apply basic Ant System
-v, --eas    apply elitist Ant System
-w, --ras    apply rank-based version of Ant System
-x, --mmas   apply MAX-MIN ant system
-y, --bwas   apply best-worst ant system
-z, --acs    apply ant colony system

-h, --help   display the help text and exit

Options --as, --eas, --ras, --mmas, --bwas, --acs,
--help don't need arguments, while all the others do
```

NetLogo : http://ccl.northwestern.edu/netlogo
procédure `setup`

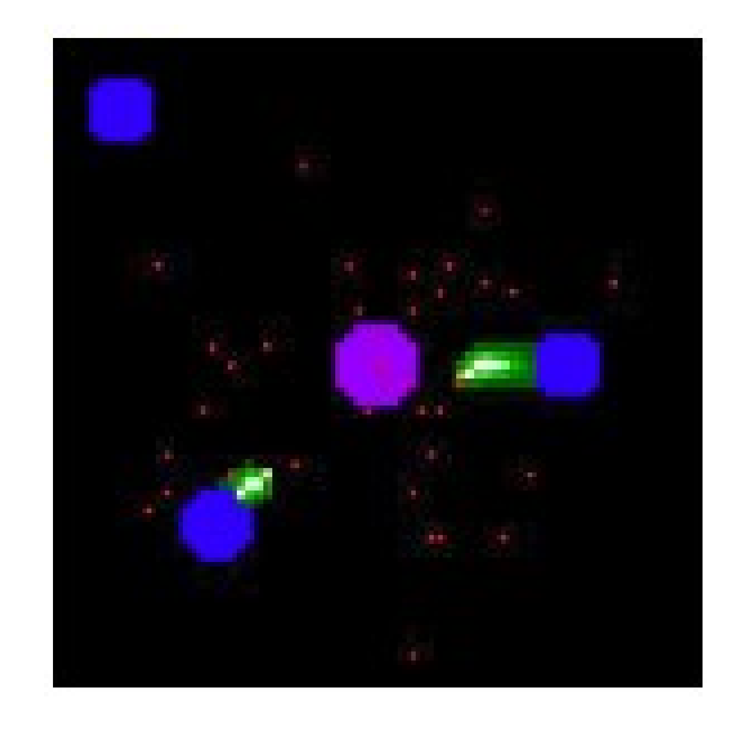
- 1 nid (violet)
- 3 sources de nourriture (bleu)
- 100 à 200 fourmis (rouge)



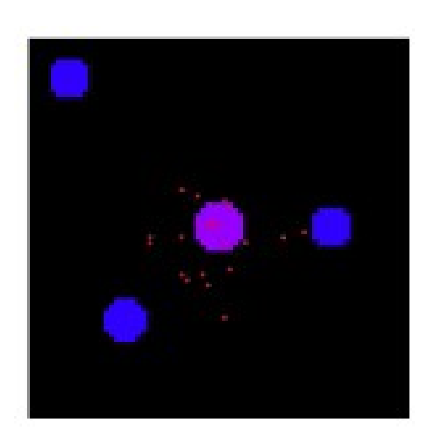
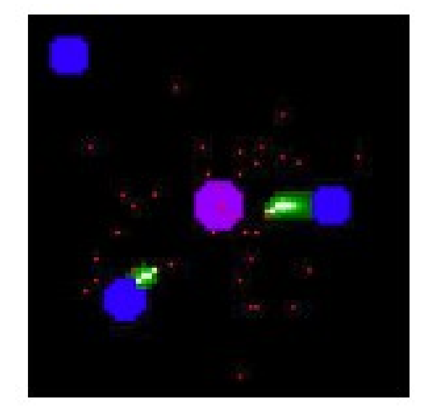
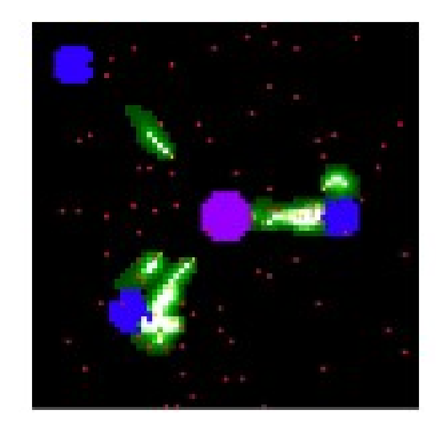
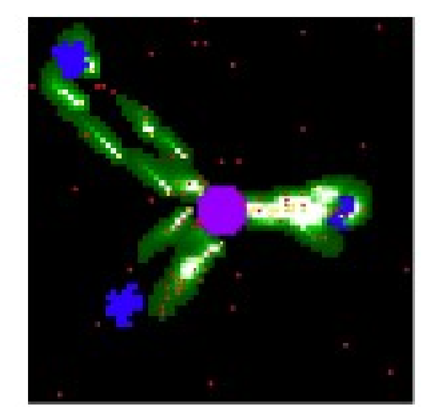
Résultat global observé

- Les ressources en nourriture sont exploitées selon leurs propre distance et attrait

- Emergence d'une décision collective (choix d'un chemin)



Modéliser & Simuler :
Un nid… et trois sources de nourritures

Comportement local des fourmis

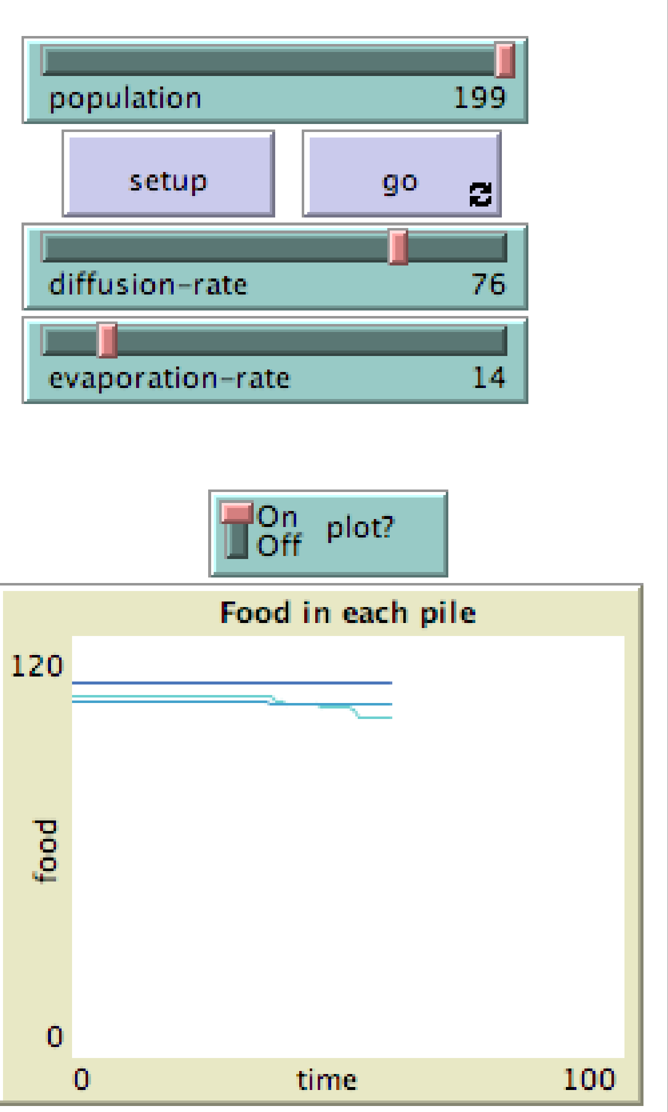
- Se déplacer au hasard
- Si trouver nourriture, la prendre et retourner au nid
- Si transporte nourriture, déposer de la phéromone
- Si ne transporte pas de nourriture, suivre les émanations de phéromone

Comportement local d'une fourmi

- Si une fourmi est sans nourriture
  - Si elle trouve de la nourriture, elle en prend, dépose des phéromones et fait demi-tour
  - Sinon, elle avance prioritairement :
    - vers la case ayant le plus de phéromone
    - vers la case sentant le plus la nourriture
    - au hasard

Comportement local d'une fourmi

- Si une fourmi possède de la nourriture
  - Si elle est dans le nid, elle la dépose et fait demi-tour
  - Sinon elle se dirige vers la case sentant le plus le nid en déposant une dose de phéromones décroissante avec l'éloignement de la source



population 199
setup
go
diffusion-rate 76
evaporation-rate 14
On
Off
plot?
Food in each pile
120
food
0
0
time
100

ticks: 58

diffusion-rate vs. evaporation-rate

```
to go
  ask turtles
  [ ifelse color = red
    [ look-for-food  ];; not carrying food? look for it
    [ return-to-nest ];; carrying food? take it back to nest
    wiggle
    fd 1 ]

  diffuse chemical (diffusion-rate / 100)

  ask patches
  [ set chemical chemical * (100 - evaporation-rate) / 100
  ;; slowly evaporate chemical
  ]
end
```

Observations!

- Les fourmis exploitent-elles les sources en série ou en // ?
- La source la plus proche est-elle exploitée en premier ?
- Y-a-t-il émergence d'un chemin stable (ie. ininterrompu) entre le nid et une source ?
- Existe-t-il une taille critique pour la colonie?

Observations!

- Déterminer le taux de fourmis « utiles » : ratio entre le nombre de fourmis qui ramènent de la nourriture et le nombre total de fourmis

- Quelle est l'influence sur la dynamique
  - du taux de diffusion des phéromones
  - du taux d'évaporation des phéromones

Pour aller plus loin …

- Essayer différents placements pour les sources de nourriture
  - Que se passe-t-il si deux sources sont à une même distance du nid ?
    - exploitation/exploitation : série ou //
  - Que se passe-t-il avec un obstacle entre le nid et une source ?
- Une fourmi utilise une « astuce » pour revenir au nid en suivant une odeur (nest-scent). Les vrais fourmis ne font pas ici ! Essayer d'implanter d'autres solutions !

Pour aller plus loin …

- Une fourmi est sensible au niveau de phéromones déposés par ses congénères (variable `chemical`) entre 0.05 et 2
  - La limite inf permet aux fourmis de ne pas rester indéfiniment sensibles aux phéromones
  - Supprimer la limite sup ! Que ce passe-t-il ? Pourquoi?

- Dans UPHILL-CHEMICAL, une fourmi suit le gradient de phéromone
  - Elle "sniff" dans trois directions, et tourne dans celle qui « sent » le plus
  - Essayer des variantes en changeant le nombre et la place des voisins

Concepts

- Règles locales simples (parallélisme)
- Emergence de « computation » collective (globale)
- Pas de « leader », pas de carte
  - actions décentralisées
- Interactions locales
  - ant ↔ ant (via l'environnement)
- Transition de phase (?seuil)
  - masse critique = # minimal de fourmis

Sources / Références

- http://www.cnrs.fr/Cnrspresse/n386/html/n386a09.htm
- E. Bonabeau, M. Dorigo and G. Theraulaz. Inspiration for optimization from social insect behaviour. Nature, Vol. 406, juillet 2000, pp. 39-42
- Bonabeau, E. & Theraulaz, G. (2000). Swarm Smarts. Scientific American, 282 (3): pp. 72-79
- E. Bonabeau, M. Dorigo and G. Theraulaz. (1999) Swarm Intelligence : From Natural to Artificial Systems. Oxford University Press

Routage par colonie de fourmis

- Les techniques de l'intelligence en essaim démontrent également toute leur puissance dans le cas de problèmes dont l'énoncé, les données ou les paramètres varient en permanence sur des échelles de temps très courtes. C'est le cas du routage dans les réseaux de communications

Routage

Lorsqu'une communication est établie entre 2 ordinateurs, le message initial est découpé en paquets de données qui circulent le long d'un réseau constitué de

1. lignes de transmission - dont les capacités de débit peuvent être très diverses et variables au cours du temps

2. de routeurs qui constituent les nœuds de ce réseau

Routage

La fonction des routeurs est de

- diriger les paquets de données vers l'un des autres routeurs du réseau et ce, jusqu'à ce que les paquets de données arrivent à leur destination finale
- tenir compte de l'importance du trafic sur les voies de communication auquel il est relié de manière à éviter l'engorgement de ces voies. Il arrive donc très souvent que des paquets de données d'un même message suivent des voies complètement différentes

Routage par colonie de fourmis Comment ça marche ?

On fait circuler en parallèle avec les paquets de données, des agents de routage, sorte de fourmis virtuelles, qui

1. analysent en temps réel l'état d'encombrement des différentes voies du réseau
2. indiquent cet état à chacun des routeurs
3. calculent le temps qu'elles mettent pour aller d'un nœud du réseau à un autre
4. marquent à l'aide de phéromone virtuelle la voie qu'elles viennent d'emprunter

Routage par colonie de fourmis
Comment ça marche ?

- Plus le délai est court, plus l'intensité du marquage est importante
- Ainsi, lorsqu'un paquet de données arrive au niveau d'un routeur donné, il aura d'autant plus de chance d'emprunter une voie que la densité de phéromone virtuelle sur cette voie sera importante
- De cette manière, le réseau s'adapte en permanence et de manière totalement décentralisée à l'activité du trafic